

BAB II TINJAUAN PUSTAKA DAN DASAR TEORI

2.1. Tinjauan Pustaka

Dalam tinjauan pustaka di bawah ini terdapat beberapa referensi yang dapat dijadikan acuan untuk penelitian, referensi tersebut adalah sebagai berikut :

Kodrat Iman Satoto (2009) melakukan penelitian mengenai sistem *mail autoresponder* yang diintegrasikan dengan Sistem Informasi Akademik. Sistem ini berfungsi untuk memberikan layanan informasi akademik melalui email secara otomatis.

Muhammad Ibnu Khoir (2014) membuat aplikasi pengolah nilai rapor pada kurikulum 2013 menggunakan *framework* Codeigniter serta MySQL sebagai teknologi basisdata. Aplikasi ini mengolah data nilai murid di SMAN 1 Prambanan.

Perbandingan antara beberapa penelitian yang digunakan sebagai tinjauan pustaka bisa dilihat pada Tabel 2.1.

Tabel 2. 1 Perbandingan Penelitian

Peneliti	Judul	Metode/ Teknologi	Bahasa Pemrograman	Interface
Kodrat Iman Satoto (2009)	<i>Autoresponder</i> Sistem Informasi Akademik	<i>Autoresponder</i>	Javamail, Java	GUI
Muhammad Ibnu Khoir	Pengolah Nilai Rapor SMAN 1 Prambanan	Framework Codeigniter	PHP, MySQL,	GUI
Sutarman	Otomatisasi Proses Baca Data Dari Email dan Simpan Ke Database Untuk Pengelolaan Virtual Account	Framework Codeigniter	PHP, MySQL	GUI

2.2. Dasar Teori

2.1.1. Email

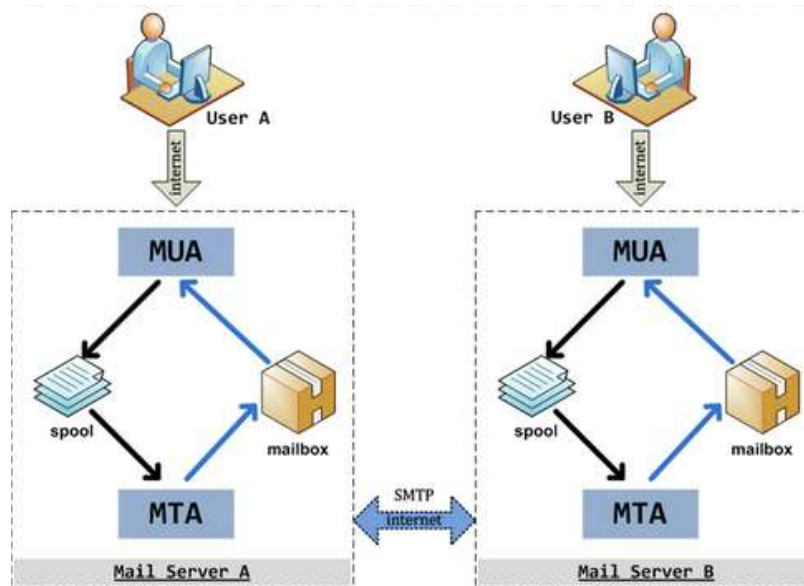
Email, layanan jaringan yang paling populer telah mengubah cara berkomunikasi masyarakat dengan kesederhanaan dan kecepatannya (Cisco, 2010). Sebelum bisa dijalankan pada sebuah komputer atau piranti lainnya, email membutuhkan beberapa aplikasi dan layanan. Dua contoh *application layer protocol* adalah *Post Office Protocol* (POP) dan *Simple Mail Transfer Protocol* (SMTP), yang berjalan dalam mode proses *client/server*. POP dan POP3 adalah *client/server protocol* yang bertugas mengirim email dari *mail server* ke *mail client*. SMTP bertugas mengirimkan email dari *client* ke *mail server* dan pengiriman email antar server. (Kusworo, 2011)

Kodrat Iman Satoto (2009) melakukan penelitian mengenai sistem *mail autoresponder* yang diintegrasikan pada Sistem Informasi Akademik. Sistem ini berfungsi untuk memberikan layanan informasi akademik melalui email secara otomatis. Sistem *autoresponder* ini dibangun dengan dua skrip yaitu skrip filter email (menggunakan Procmail) untuk memilih pesan informasi yang valid dan skrip untuk menjawab secara otomatis (menggunakan PHP)

2.1.2. Simple Mail Transfer Protocol

Simple Mail Transfer Protocol (SMTP) merupakan protokol yang akan melayani aplikasi elektronik mail atau email. Email terkenal karena memberikan cara yang mudah dan cepat dalam mengirimkan informasi. Selain itu email ini dapat

menangani file yang kecil maupun file yang besar. Proses pengiriman email dapat dilihat pada gambar di bawah ini.



Gambar 2. 1 Proses Pengiriman Email

Pemakai di terminalnya berhubungan dengan *user agent* (UA). Beberapa agent email yang populer antara lain adalah Pine, Pegasus dan Eudora. Pertukaran *mail* menggunakan TCP dilakukan oleh *Message Transfer Agent* (MTA). MTA yang paling umum untuk UNIX adalah sendmail. Pemakai awam pada umumnya tidak berhubungan dengan MTA ini. (Kusworo, 2011)

2.1.3. Post Office Mail

Post Office Mail merupakan protokol yang digunakan untuk mengelola email. POP mempermudah seseorang dalam mendapatkan *mail* dari sebuah *mail server* tanpa perlu koneksi yang lama dengan internet yang tentu saja memakan biaya. Menurut Widia (Onno W Purbo, 2000).

Standar POP untuk internet saat ini adalah POP3 (POP versi 3) protokol ini yang mengijinkan *client* mengakses *mail* secara dinamis. Sama seperti halnya dengan pegawai kantor pos, POP3 menerima dan memegang email bagi individu sampai mereka mengambilnya, dan sebanyak kantor pos tidak membuat salinan dari surat yang diterimanya. (Onno W Purbo, 2000).

Dalam versi POP3 sebelumnya, ketika sebuah email didownload individu dari server ke dalam program email mereka, tidak ada lebih banyak salinan email pada server POP, dalam artian email yang sudah didownload tidak akan ada lagi atau tidak bisa didownload kembali karena email tersebut secara otomatis terhapus. (Onno W Purbo, 2000).

Versi POP3 saat ini berkembang pesat sehingga beberapa administrator dapat mengkonfigurasi protokol untuk email di server pada jangka waktu tertentu, sehingga memungkinkan seseorang untuk mendownload sebanyak yang mereka inginkan, dan juga terdapat pengaturan pada *client* yang memberikan pilihan untuk tetap menyimpan data server. (Onno W Purbo, 2000).

Beberapa kelebihan yang dimiliki POP (*Post Office Protocol*) adalah :

1. Email disimpan pada penyimpanan lokal, sehingga dapat diakses setiap saat, bahkan tanpa koneksi internet.
2. Koneksi internet hanya dibutuhkan saat mengirimkan dan menerima email.
3. Mengurangi beban pada kapasitas penyimpanan server, karena data disimpan di komputer lokal.
4. Terdapat pilihan untuk tetap menyimpan email di server.

5. Dapat menggabungkan banyak akun mail pada 1 email *client*, dengan menggunakan folder inbox yang sama.

2.1.4. Internet Message Acces Protocol

IMAP (*Internet Message Acces Protocol*) merupakan protokol standar untuk mengakses dan mengambil email dari server. IMAP memungkinkan pengguna memilih pesan email yang akan diambil, membuat folder di server, mencari pesan email tertentu, bahkan menghapus pesan email yang ada. Kemampuan ini jauh lebih baik dari pada protokol POP3 yang hanya diperbolehkan mengambil atau mengunduh semua pesan yang ada tanpa terkecuali. (Onno W Purbo, 2000).

Beberapa kelebihan yang dimiliki oleh IMAP adalah sebagai berikut :

1. Email disimpan pada remote server, yang berarti email dapat diakses secara bersamaan dari beberapa lokasi sekaligus.
2. Koneksi internet dibutuhkan untuk mengakses email.
3. Menghemat kapasitas penyimpanan komputer lokal.
4. Waktu *reload* email lebih cepat dan sinkronisasi antara email server dan komputer remote akan selalu terjadi secara otomatis dan melakukan aktivitas di komputer.
5. Terdapat pilihan untuk menyimpan email ke penyimpanan local.

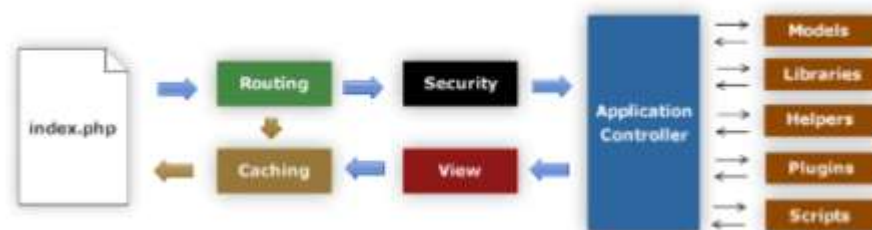
2.1.5. Codeigniter

Codeigniter merupakan *open source* berbasis MVC (*Model View Controller*). Untuk membangun aplikasi web dinamis dengan cepat dan mudah. Codeigniter

memiliki desain dan struktur file yang sederhana, didukung dengan dokumentasi yang lengkap sehingga *framework* ini lebih mudah dipelajari. (Septian, 2011)

Codeigniter ini memungkinkan para pengembang untuk menggunakan *framework* secara parsial atau secara keseluruhan. Artinya bahwa codeigniter masih memberi kebebasan kepada para pengembang untuk menulis bagian – bagian kode tertentu di dalam aplikasi menggunakan cara konvensional atau dengan sintak umum di dalam PHP, tidak harus menggunakan aturan penulisan kode di Codeigniter (Septian, 2011).

Adapun alur dari aplikasi yang ditulis menggunakan Codeigniter adalah sebagai berikut :



Gambar 2. 2 Alur Penggunaan Codeigniter

File index.php, berfungsi sebagai *front controller*, menginisialisasi *resource* utama yang dibutuhkan untuk menjalankan Codeigniter.

1. *Router*, memeriksa HTTP *request* untuk menentukan apa yang harus dilakukan.
2. Jika file *cache* ada, dikirim langsung ke browser, melewati eksekusi sistem normal.
3. Keamanan, sebelum *controller* aplikasi dimuat, HTTP *request* dan setiap data pengguna yang dikirim dan disaring dahulu untuk keamanan.

4. *Controller* memuat model, *library* utama, *helper* dan setiap *resource* lainnya yang diperlukan untuk memproses permintaan khusus.
5. *View*, proses render kemudian dikirim ke web browser agar dapat dilihat. Jika caching diaktifkan, *view* dicache terlebih dahulu sehingga pada permintaan berikutnya dapat dilayani.

2.1.6. PHP

Menurut (Rasmus Lerdorf, 1995) PHP adalah bahasa pemrograman *script* yang paling banyak dipakai saat ini. PHP banyak dipakai untuk memprogram situs *web* dinamis, walaupun tidak tertutup kemungkinan untuk digunakan pada pemakaian lain. PHP merupakan bahasa berbentuk *script* yang ditempatkan dalam *server* dan diproses di *server*. Hasil yang dikirim ke *client*, tempat pemakai menggunakan *browser* (Kadir, 2008:2).

Secara khusus, PHP dirancang untuk membentuk aplikasi dinamis. Artinya PHP dapat membentuk suatu tampilan berdasarkan permintaan terkini. Misalnya menampilkan isi *database* ke halaman *web*. Pada prinsipnya PHP mempunyai fungsi yang sama dengan *script – script* seperti ASP (*Active Server Page*), *Cold Fusion*, ataupun *Perl*. Namun perlu diketahui bahwa PHP sebenarnya biasa dipakai secara *command line*. Artinya, *script* PHP dapat dijalankan tanpa melibatkan *web server* maupun *browser*.

PHP diciptakan oleh Rasmus Lerdorf, seorang pemrogram C. Rasmus Lerdorf membuat sejumlah *script* Perl yang dapat mengamati siapa saja yang melihat – lihat daftar riwayat hidupnya, yakni pada tahun 1994. *Script – script* ini selanjutnya dikemas menjadi *tool* yang disebut “Personal Home Page”. Paket ini yang

menjadi cikal bakal PHP. Pada tahun 1995, Rasmus menciptakan PHP/FI Versi 2. Pada versi inilah pemrograman data menempelkan kode terstruktur di dalam tag HTML. Kode PHP juga dapat berkomunikasi dengan *database* dan melakukan perhitungan – perhitungan yang kompleks.

Konsep kerja PHP diawali dengan permintaan suatu halaman *web* oleh *browser*. Berdasarkan URL (Uniform Resource Location) atau dikelas dengan sebutan alamat internet, *browser* mendapatkan alamat dari *web server*, mengidentifikasi halaman yang dikehendaki, menyampaikan segala informasi yang dibutuhkan oleh *web server*. Ketika berkas PHP yang diminta diperoleh *web server*, isinya segera dikirim ke mesin PHP dan mesin inilah yang memproses dan memberikan hasil (berupa kode HTML) ke *web sever*. Selanjutnya *web server* menyampaikan ke *client*.

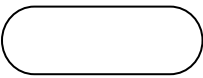

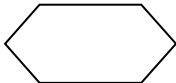
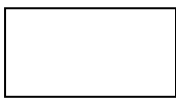
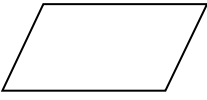

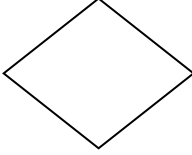
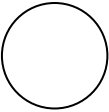
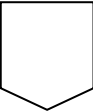
2.1.7. Flowchart

Bagan alir (Flowchart) adalah bagan (chart) yang menunjukkan hasil (flow) di dalam program atau prosedur sistem secara logika. Vagan alir digunakan terutama untuk alat bantu komunikasi dan untuk dokumentasi (Jogiyanto, 2005).

Flowchart adalah bagan – bagan yang mempunyai arus yang menggambarkan langkah – langkah penyelesaian suatu masalah flowchart merupakan cara penyajian dari suatu algoritma (Ladjamuddin, 2006).

Berikut merupakan simbol – simbol dan keterangan dari flowchart :

Tabel 2. 2 Simbol Flowchart

Simbol	Nama	Keterangan
	Terminator	Permulaan atau akhir program
	Flow Line	Arah aliran program
	Preparation	Proses inisialisasi atau pemberian harga awal
	Proses	Proses Perhitungan atau proses pengolahan data
	Input/Output Data	Proses input atau output data, parameter, informasi
	Predefined Process	Permulaan sub program atau proses menjalankan sub program
	Decision	Perbandingan pernyataan , penyeleksian data yang memberikan pilihan untuk langkah selanjutnya.
	On Page Connector	Penghubung bagian – bagian flowchart yang ada pada satu halaman
	Off Page Connector	Penghubug bagian – bagian flowchart yang ada pada halaman berbeda.

2.1.8. MySQL

Menurut Aprillita (2013:2), MySQL adalah salah satu perangkat lunak *Database Manegement System* (DBMS) yang sering digunakan saat ini, yang didistribusikan secara gratis di bawah lisensi GPL (*General Public Licence*). Sehingga setiap orang mudah untuk mendapatkan dan bebas untuk menggunakan MySQL. MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (Bahasa Inggris : *database management system*) atau DBMS yang multithread, multiuser, dengan sekitar 6 juta instalasi di seluruh dunia.

Tidak seperti *Apache* yang merupakan *software* yang dikembangkan oleh komunitas umum, hak cipta untuk kode sumber yang dimiliki oleh penulisnya masing – masing. MySQL dimiliki dan disponsori oleh sebuah perusahaan komersial Swedia yaitu MySQL AB. MySQL AB memegang penuh hak cipta hampir atas semua kode sumbernya. Yang mendirikan MySQL adalah dua orang swedia dan satu orang Finlandia yaitu David Axmark, Alan Larsson, Muchael “Monty” Widenius (Solichin, 2010:10).

Dalam MySQL terdapat beberapa perintah dasar yang akan sering digunakan dalam pengelolaan basis data, perintah – perintah tersebut dibagi atas tiga kelompok yaitu sebagai berikut :

A. Data Definition Language (DDL)

DDL merupakan kumpulan perintah SQL yang digunakan untuk membuat, mengubah, dan menghapus struktur dan definisi metadata dari objek – objek database. Secara umum, DDL yang digunakan adalah Create untuk membuat objek baru, Use untuk menggunakan objek, Alter untuk mengubah objek yang sudah ada,

dan Drop untuk menghapus objek. DDL biasaya digunakan oleh administrator basis data dalam pembuatan sebuah aplikasi basis data.

B. Data Manipulation Language (DML)

DML merupakan kumpulan perintah SQL yang digunakan untuk proses pengolahan isi data di dalam tabel seperti memasukkan, mengubah dan menghapus isi data dan tidak terkait dengan perubahan struktur dan definisi tipe data dari objek database. DML digunakan untuk memanipulasi data yang ada dalam suatu tabel. Perintah yang umum digunakan adalah Select untuk menampilkan data, Insert untuk menambah data baru, Update untuk mengubah data yang sudah ada, dan Delete untuk menghapus data.

C. Data Control Language (DCL)

DCL merupakan perintah – perintah yang dapat digunakan untuk menjaga keamanan basis data, perintah tersebut dapat dipakai untuk menentukan akses basis data hanya dapat dilakukan oleh orang – orang tertentu dan dengan macam akses yang dibatasi pula. DCL digunakan untuk melakukan pengontrollan data dan *server database*. Perintah yang umum digunakan adalah Grant untuk memberikan hak/ijin akses oleh administrator (pemilik utama) *server* kepada user (pengguna biasa) dan Revoke untuk menghilangkan atau mencabut hak akses yang telah diberikan kepada *user* oleh *administrator*.